An Adaptive Non-Strobed Sensing Scheme for Nano-scale SRAMs

Sudhanshu Khanna, Electrical and Computer Engineering, University of Virginia

ABSTRACT

To achieve high density, SRAMs use aggressively small bit-cells. As a result the small sized bit-cells dominate SRAM delay. Local variation in nano-scale technologies compounds the problem as the near minimum sized transistors of bit-cells are worst hit by such variation. The conventional solution is to estimate the variation and to include enough margins in SRAM timing. This paper proposes a novel Adaptive Non-strobed Sensing scheme (ANS) scheme where the delay of the SRAM varies with the bit-cell being read. As a result the SRAM is fast most of the times, but is slow while reading the few slow bit-cells. A shadow circuit detects the timing violation and issues a "data invalid" signal to the processor along with the correct data in the *next* cycle. The scheme improves average SRAM delay by 12%.

1. INTRODUCTION

SRAMs are a critical component in most modern digital systems ranging. To lower area and hence cost per chip, significant amount of effort is put into lowering bit-cell area. At the same time, technology scaling lowers area by half every two years further reducing cost. However, as technology scales, local variation in transistor parameters increases. Further, small transistors like those used in SRAM bit-cells are worst affected by this variation [1]. Thus, SRAM delay is limited by the weakest bit-cell. Variation in bit-cell read current is near-gaussian, but with multi-million bit-cells on a die, the tail of the distribution is more than 6 sigma away from the mean. Thus the worst case delay of a SRAM is often 2-3 higher than the mean delay.

In this paper, we present an Adaptive Non-strobed Sensing scheme (ANS) which reacts to the variation in *each bit-cell* rather than using a worst case timing approach. As a result the SRAM is fast most of the times, but is slow while reading the few slow bit-cells that lie at the tail of the distribution. A shadow circuit detects the timing violation and issues a data invalid signal to the processor in the *next* cycle. Correct data is made available along with the data invalid signal resulting in single cycle penalty in the rare event of a slow read. Thus over a large number of reads, the memory has lower average delay. Conventional (strobed) sense amplifier topologies [2] can not realize this adaptive behavior because they use a fixed timing pulse to enable the sense amplifier based on the studied worst case variation. We show circuit and architectural techniques that help achieve this adaptive behavior and make the scheme usable in synchronous digital systems. Drawing from our designs and analysis, this paper makes the following key contributions:

- Quantifies the delay variation of non-strobed sensing schemes.
- Shows that conventional (strobed) sensing can only be designed with a worst case timing scheme and thus can not be used to design an SRAM whose delay depends on the particular bit-cell being read.
- Demonstrates a new sensing scheme called ANS which uses a non-strobed sense amplifier along with circuit and architectural techniques to enable local-variation-adaptive SRAM delay. The technique improves SRAM read delay by 12%.

This paper is organized in the following manner. Section 2 analyses conventional (strobed) and non-strobed sensing schemes. Section 3 describes the Adaptive Non-strobed Sensing scheme and the circuit and architectural techniques that make it useful for synchronous digital systems. Section 4 concludes the paper.

2. TIMING IN STROBED AND NON-STROBED SENSING SCHEMES

In this section we analyze the operation of conventional (strobed) and non-strobed sensing schemes. We would refer to conventional sensing as strobed sensing from here on. A strobed sensing scheme works as shown in Figure 1. SRAM sub-blocks use timing signals generated by a timing block using the system clock and delay elements. A delay T_{DEC} after the rising clock edge the SRAM timing block generates the word-line enable (WLE) pulse. This delay is set by the row decoder delay or the bit-line (BL) pre-charge delay, whichever is higher. Then, a delay T_{BC} after the WL enable rising edge, the sense amplifier enable (SAE) pulse arrives, as shown in Figure 1. T_{BC} is set such that the weakest bit-cell has enough time to pull the BL low enough that the voltage difference between BL and BLB is greater than the sense amplifier (SA) offset. It is this set delay, T_{BC}, which makes the strobed sensing scheme invariant to bit-cell strength. Even if a stronger bit-cell is being read, the SRAM read delay will be fixed at $T_{DEC} + T_{BC} + T_{SA}$. T_{SA} is the SA delay. T_{SA} and T_{DEC} are usually much smaller than T_{BC}.

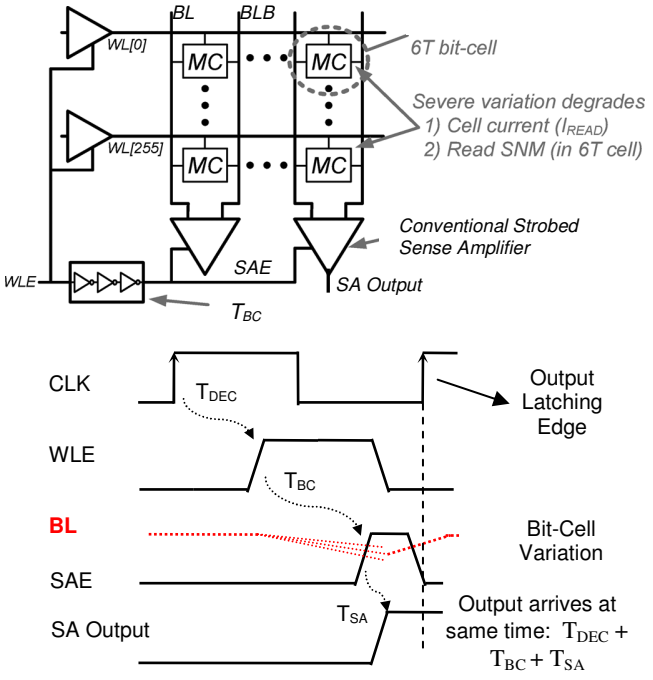


Figure 1: Strobed SRAM timing [2] and schematic [3]; BL signal shows variation due to bit-cell strength variation. Timing of SAE is set to sense the worst case BL signal. Thus, SA Output is limited by arrival of SAE rising edge and making SRAM delay the same, irrespective of bit-cell variation.

The timing of a non-strobed sensing scheme is significantly different. In this paper we use a single ended non-strobed regenerative sense amplifier (NSR-SA) from [3], however the concepts and results described in our work can be applied to other implementations of single ended or differential non-strobed sense amplifiers as well.

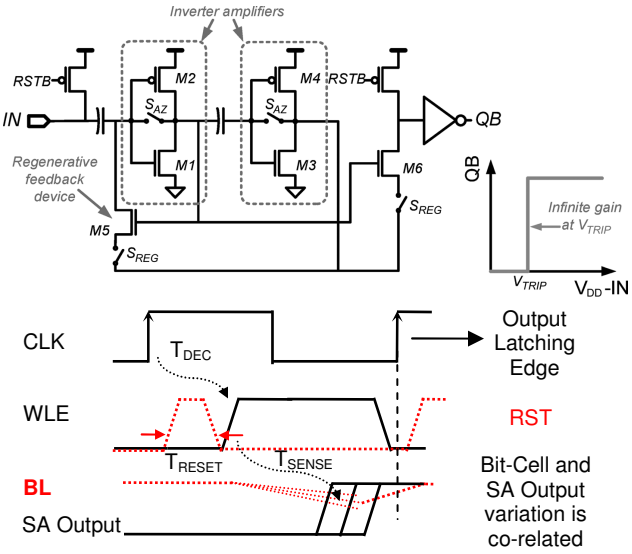


Figure 2: Non-Strobed Sense Amp (NSR-SA) [3] schematic and timing; RSTB is inverted RST. S_{AZ} switches turn on when RST=1, S_{REG} turn on when RST=0. BL signal shows variation due to variation in bit-cell. SA Output is dependent on bit-cell variation. Clock period must be set according to the worst case bit-cell, so that worst-case SA Output can be latched.

The NSR-SA timing [3] is shown in Figure 2. The NSR-SA needs a resetting pulse (of width T_{RESET}) just before the WLE turns on. This resetting pulse biases the two inverters M1-M2 and M3-M4 at their respective switching thresholds by turning on switches S_{AZ}. The reset can happen in parallel with row decode, and hence doesn't consume any extra delay. The reset pulse ends just before the WLE pulse arrives. As soon as the WLE goes high the bit-line starts drooping in case the bit-cell contains a zero. This droop gets coupled to the inverter M1-M2 input through the input capacitor, and then ripples to the output of the sense amplifier. The output of the sense amplifier reaches a low much faster than the bit-line droop because of inverter gain and the regenerative feedback provided by M5, thereby realizing sense amplifier behavior. Thus, T_{SENSE} after the WLE pulse turn on a valid output is available. T_{SENSE} depends heavily on the bit-cell being read, and to a small extent on the strength of the transistors in the sense amplifier. The WLE pulse remains high till a valid output develops on the NSR-SA [3], and is thus set by the worst case bit-cell. There are two important differences from the strobed case. First, there is no SAE pulse required. Second, because T_{SENSE} depends on the bit-cell strength, a stronger bit-cell bring read would result in a lower T_{SENSE}. Finally the output of the NSR-SA is latched at the next rising edge of the clock. It is important to notice here that the clock period, and thus the time when the clock edge latches the NSR-SA output must be after the slowest T_{SENSE}. Thus, when used in the manner showed in Figure 2, NSR-SA delay is also limited by the worst case bit-cell. In the next section we show how the NSR-SA can be used with a modified timing to implement Adaptive Non-strobed Sensing.

3. ADAPTIVE NON-STROBED SENSING

Adaptive Non-strobed Sensing (ANS) is a sensing method that allows a memory to have a delay that follows bit-cell variation. This section describes how we implement this idea, and reports the performance benefits in a commercial 45nm technology node.

As shown in Figure 1, a strobed sensing scheme would give the same SRAM delay for all read operations, making it difficult to use strobed sense amplifiers for ANS. Non-strobed sensing helps us go one step towards implementing ANS because NSR-SA timing shown in Figure 2 has a bit-cell access delay T_{SENSE} which is dependent on the bit-cell strength. However, both the WLE pulse width and the instant when the NSR-SA output is latched (and thus the clock period) are dependent on the worst case T_{SENSE}, making it difficult to use the timing in Figure 2 for ANS.

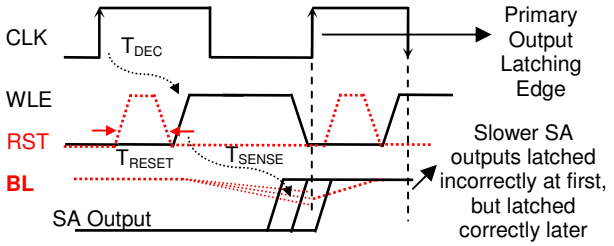


Figure 3: ANS with modified NSR-SA timing; T_{SENSE} and clock period are shorter than in Figure 2. Clock period is NOT set according to the worst case bit-cell, and thus incorrectly latches slower SA outputs.

WLE pulse width is important because the SRAM can not be prepared for the next cycle pre-charge before WLE goes down, thus impacting memory throughput. Clock period impacts both latency and throughput.

The modified timing scheme in Figure 3 helps overcome both the WLE and clock period limitations. First, we latch the output of the NSR-SA before the worst case T_{SENSE}, meaning that a few read operations latch the wrong data. However, we allow the sense amplifier to still work on developing the correct output, and we latch this correct output later. Second, at the same early latching time, we also turn off the WLE pulse, allowing the SRAM to start preparing for the next cycle pre-charge. This truncated WLE is a significant departure from the timing in Figure 1, but through transistor sizing we ensure that even though the WLE is turned off before T_{SENSE}, the correct data is developed on the sense amplifier output across global and local process variations. The WLE needs to be enough such that inverter M1-M2 has enough droop below its switching threshold to get the regeneration process into action. A larger WLE pulse would make the sense amplifier delay lower, but a slower sense amplifier does not hurt us because in case of a slow read, we check the sense amplifier output a half clock cycle later, which is enough time for the regeneration circuit to develop correct output even in absence of the higher droop on the sense amplifier input.

In this way, the SRAM works at the faster clock speed, but is prone to error once in a while. However, the error can be detected by implementing a shadow circuit that compares the output that was previously latched and the output of the sense amplifier half a cycle later, as shown in figure 3 and 4. In case the two values differ, a data invalid signal is issued, instructing the processor to stall for a cycle, and use the updated data as shown in Figure 4. The shadow circuit is a XOR gate for each data bit following by a

multi-input OR gate that issues a data invalid signal if any of the XOR gates gives a high output. A high level block diagram showing ANS being used in a system is shown in figure 4.

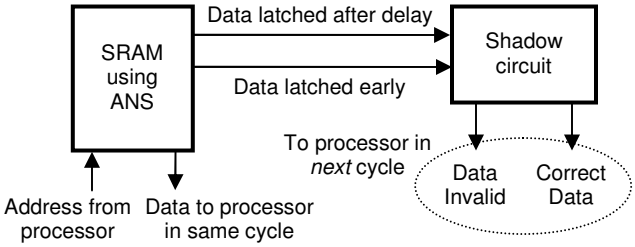
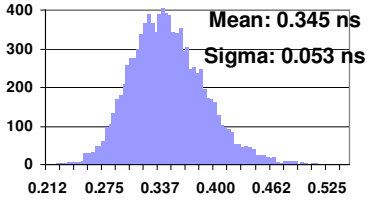


Figure 4: A SRAM using ANS along with the shadow circuit that detects incorrect reads. The (previous) data invalid signal and the (previous) correct data are made available to the processor available within the next cycle.

Figure 5 shows the distribution of T_{SENSE} (over 10K MC iterations) for the NSR-SA. With this distribution, to achieve ~99.5% read success for a bit-cell, T_{SENSE} must be lower than 2.8 standard deviations above the mean, which corresponds to 0.49ns. This means that any bit-cell that takes more than 0.49ns to be read would be read incorrectly. Note that depending on a particular implementation and the variation exhibited by the technology node in use, we may choose a threshold different from 99.5%. The 6-sigma worst case bit-cell needs a T_{SENSE} of 0.66ns to be read correctly, which would correspond to an SRAM using worst case timing. The benefit of ANS comes from this difference, which is 170ps (0.66ns-0.49ns, or 6 sigma – 2.8 sigma). Taking the example of a memory with 16-bit word, a success rate of 99.5% per bit-cell translates to a 92.3% success rate for a 16-bit read operation (0.995^16=0.923). To find the average delay we have to factor in the fact that some reads in ANS will be 2 cycles long. Assuming that other SRAM components (decoder, WL driver etc.) have a collective delay of 0.25ns, the average SRAM delay is 0.8ns (0.923*(0.49ns+0.25ns) + (1-0.923)*2*(0.49ns+0.25ns)). A SRAM not using ANS would have a delay limited by the 6 sigma worst case T_{SENSE}, and would be 0.91ns (=0.66ns + 0.25ns), assuming the same 0.25ns delay for other SRAM components. *Thus, using ANS results in a 12% performance benefit in the average case.* Also, the different between $T_{SENSE@0.995}$ and $T_{SENSE@WC}$ is 3.2 standard deviations (6 sigma – 2.8 sigma), which increases with variation in T_{SENSE}. Thus, the benefit of ANS would increase with technology scaling. Figure 5 also shows the benefit of ANS as T_{SENSE} sigma increases, mimicking the impact technology scaling.



Sigma (ns)	% Benefit
0.05	12
0.10	21
0.15	27
0.20	31

Figure 5: NSR-SA T_{SENSE} distribution over 10K MC runs in a commercial 45nm node. The table shows the benefit of using ANS with increased T_{SENSE} standard deviation.

A circuit issue in implementing ANS is in reading a previous read data value while the next read operation is going on. The output of the NSR-SA is valid only till the reset pulse (of the next read operation) arrives. In the ANS scheme we intend to read the sense amplifier output at the falling edge following a read operation to determine if the previously latched data was correct or not. However by this time the next read operation would have started, as shown in figure 3. Also, as shown in Figure 3, by the time the falling edge arrives the next reset pulse already corrupts the data. As a solution to this problem we propose a Sense Amplifier Alternation (SAA) scheme shown in figure 5. The scheme involves having two NSR-SAs per column and alternating between them during consecutive read accesses. This way, the NSR-SA has an entire cycle to develop the output in case of a slow read. This alternation scheme can also be used standalone (without ANS) in memory designs using the NSR-SA to reduce the impact of high reset time on SRAM delay. Adding an extra sense amplifier is a small area overhead in the overall SRAM context, but getting rid of the reset time is essential in our scheme, and is a good area-performance trade-off in standalone SRAMs working at low supply voltages, where reset time shoots up sharply. The reason it promises to be a good area-performance overhead is that any sense amplifier contributes significantly to performance but only marginally to overall SRAM area.

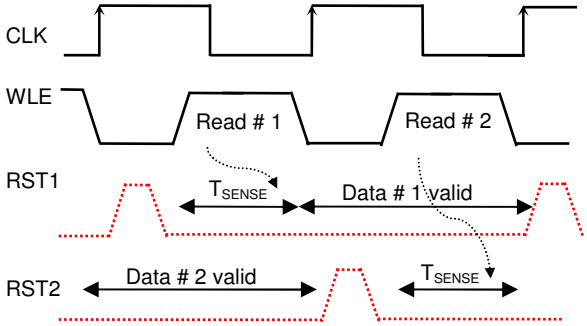


Figure 5: SAA scheme. Having two NSR-SAs per column and using them for alternate read operations, allows reading each NSR-SA for an entire cycle after a read access.

4. CONCLUSION

Local variation is a growing concern in nano-scale technology nodes. SRAMs are most vulnerable to local variation because they use large numbers of small sized transistors. Conventional strobed sensing schemes estimate the worst case bit-cell and include enough margins in SRAM timing to read such outlying bit-cells. We propose a novel sensing scheme that adapts to individual bit-cell strength and aims to correctly read majority of the bit-cells. For the minority of the reads that fail, a shadow circuit detects the fail, and sends a data invalid signal to the processor along with the correct data in the next cycle. Non-strobed sensing with a novel timing methodology and a sense amplifier alternation to increase sense amplifier output-valid time help implement this Adaptive Non-strobed Sensing scheme (ANS). ANS helps reduce SRAM delay by 12%. We also show that with increased variation caused by continuing technology scaling, the benefit provided by ANS increases. Finally, the concept is shown using simulations for a 45nm SRAM but is applicable to any memory in general.

5. REFERENCES

[1] Miyamura, M. "SRAM critical yield evaluation…", VLSI Tech 2006

[2] Wicht, B. "Yield and speed optimization of a latch-type voltage sense amplifier", JSSC July 2004

[3] Verma, N. "A High-Density 45nm SRAM Using Small-Signal Non-Strobed Regenerative Sensing", ISSCC 2008